**Problem Statement**

An **object store** accessible via REST APIs.

Project Description:-

This is an REST Web Service project in which Data can be accessed or modified via REST API of jersey from the database. A user can request through single URL to the server and server will send the response with HTTP status and data will be JSON format.

Tools for Implementation of Project:-

* Eclipse
* Postman (which is a REST client where a user can send request (GET, POST, PUT, DELETE) through URL)

Implementation of Project:-

First, create a server run time for Tomcat 6.0 on Eclipse. This is the Web container for your RESTful Web application. Then create a dynamic Web application named "Jersey," and specify the target run time to be Tomcat 6.0.

Now that you have set up the environment you are ready to develop your first REST service, which simply says "Hello" to the client.

To do this, you need to direct all the REST requests to the Jersey container by defining a servlet dispatcher in the application's web.xml file. Besides declaring the Jersey servlet, it also defines an initialization parameter indicating the Java package that contains the resources.

Project Configuration:-

In Configuration, we provide the connection of the database for storing, accessing, and updating the data.

For the database connectivity I used the My SQL jdbc connection and provide connectivity in Data Source Class. Here you can provide your database path to connect with database. My Database path is

Connection con=DriverManager.*getConnection*("jdbc:mysql://localhost:9001/frshep","root","8778");

Here the Database format is:-

* In My database, name of the database is “dharam” and table name is “USER\_DETAILS”
* There are two tables in this database first one is “name” and second is “pass” in which a user can give the data in this format.

Table name: USER\_DETAILS

|  |  |
| --- | --- |
| Name | pass |
|  |  |
|  |  |

Communicating URL’s:

* <http://localhost:8080/jersey/webapi/register>

This URL is will get the user details name and pass for the registering the user and when user already exist it will give response status .

If there is no any user in the database it will insert the user in the table.

* <http://localhost:8080/jersey/webapi/retrive>

In this URL will take the user data and then validate it, if it exist in the database it will provide the user detail in JSON format.

If there is no any user of same name it will give the message “User does not exist”.

* <http://localhost:8080/jersey/webapi/update>

In this URL, it will take the user detail and validate it, if the user details are correct then delete/update the row of the table and the give the message to user “Successfully updated”.